Aqui aprenderei o JS, fazendo anotações e organizando meu tempo, para a conclusão desse curso. Caso eu tenha uma dúvida estará anotado tudo o que eu precisar inclusive informações inúteis. Utilizarei como fonte principal os vídeos do “curso em vídeo”.

**Legenda:** Frases que estão com “// e a linha pintada de roxo” são comentários meus.

**Aula 01: Visualizado dia 28/09/2020**

**O que é cliente e servidor?**

R:. Cliente é basicamente aquele que consome, seria toda a infraestrutura de uma pessoa

Servidor seria de onde é distribuído os links, as informações.

**HTML:** É onde você coloca a informação (jornalista), conteúdo

**CSS:** Seria o Designer (designer)

**JS:** É onde é modelado a informação (programador), interatividade do lado do cliente

**Aula 02: Visualizado dia 28/09/2020**

4 exemplos que usam JS: Google, Netflix, LinkedIn, Youtube

Nome JavaScript: Foi criado por que em 1995 a palavra mais falada era Java (famosa jogada de marketing)

Datas de Acontecimento até o nascimento do JavaScript.

* 1993: Nascimento do WWW;
* 1994: Nascimento do Netscape (empresa que mexia com internet)
* 1995: Surgimento do nome JavaScript, criado pela Netscape
* 1997: Netscape padronizou a linguagem, Ecmascript significa então a versão padronizada de javascript.
* 2002: Netscape faliu, Nascimento do Firefox.
* 2008: Criação do Google Chrome.
* 2009: criação do V8, seria o motor que roda o Google Chrome.
* 2010: nodeJS= criado e roda o JavaScript fora do navegador

**Aula 03: Visualizado dia 01/10/2020**

Ecmascript = Versão Padronizada do JS.

Bibliografia: JavaScript o guia definitivo

JavaScript Guia do Programador

Guia de Referência da Mozilla: developer.mozilla.org

Guia de Referência Ecma: ecmainternational.org(standat ecma 262)

Para aprender utilizarei o Nodejs e VSCode.

**Aula 04: Visualizado dia 04/10/2020**

***Começando a criar o meu primeiro script - 04***

**<Head> =** Lugar onde é definido como será o site, configurações básicas(ex: se ele vai ser compatível com o Windows, ou se ele vai poder colocar pontuação)

nesse lugar também é definido a parte CSS(seria o style) do html

**<body> =** defino o conteúdo do site

Neste lugar também é programado o JS (script)

Alguns comandos:

* window.alert //seria basicamente para a página mostrar uma mensagem
* window.confirm //seria basicamente para a página mostrar algo para você confirmar
* window.prompt //seria basicamente para a página mostrar algo para você escrever antes de mostrar o conteúdo da página

MODULO **A** FINALIZADO!!!

INICIANDO O MODULO **B**!!!

**Aula 05: Visualizado dia 04/10/2020**

***Variáveis e Tipos Primitivos***

**"="** : Recebe

**a1 = null** (vazio, vai ficar sem nada)

No JavaScript moderno, além de utilizar a palavra *"var"* para variável, também podemos usar a palavra *“let”* e *“const”*.

**Atribuição:** Quando uma variável recebe alguma coisa (seja números ou palavras)

**Identificador:** Nome de uma variável.

- Podem começar com $ ou \_

- Não podem começar com números

- É possível usar letras ou números

- É possível usar acentos e símbolos

- Não podem conter espaços

- Não podem ser palavras reservadas (ou seja, não pode utilizar uma variável com o nome de algum comando)

**Data types:**

typeof - Seria um comando para descobrir o tipo da variável. (geralmente usado no nodeJS)

1. number:
   * 1. Infinity
     2. Nan
2. string
3. Boolean
4. null
5. undefined
6. object:
   * 1. Array
7. Function

**Aula 06: Visualizado dia 08/10/2020**

***Tratamento de dados***

Quando você utiliza o símbolo "+", no JS ele significa concatenação, ou seja, ele junta frases, a exemplo:

"mensagem" + dado, dessa forma a mensagem e o dado estarão conectados.

porém também serve para adição

Exemplo de como funciona ambos.

(number) + (number) para adição

(string) + (string) para concatenação

Para converter uma string no JS em números utilize:

Number.parseInt(n) // Inteiro

Number.parseFloat(n) // Real

Também pode usar,

Number(n) //Nesse caso ele converte para os dois, o JS se vira para entender se é real ou inteiro.

Agora Para fazer a conversão para String é necessário usar um dos seguintes comandos:

String(n)

n.toString()

**Formatando Strings:**

var s = 'JavaScript'

*'Estou aprendendo s'* //não faz interpolação

*'Estou aprendendo ' + s* //Usa concatenação

*`Eu estou aprendendo ${s}`* //usa template string

Utiliza a crase para concatenar mais fácil

s.length //quantos caracteres a string tem

s.toUpperCase() //tudo para "MAIUSCULAS"

s.toLowerCase() //tudo para "minúsculas"

**Novo comando:**

document.write() //Esse comando escreve na tela, na área "do site(dentro do site, ou seja, ele não abre uma caixa de mensagem)".

**Formatando Números:**

var n1 = 1543.5

n1 // exibe o número normal.

n1.toFixed(2) //Exibe o número com 2 casas decimais.

n1.toFixed(2).replace('.',',') //Troca o ponto pela virgula.

n1.toLocaleString('pt-BR',{style:'currency', currency:'BRL'}) // Transforma o valor em valor real por causa do BRL.

//localeString - Localizar String em pt-br.

//currency – moeda.

**Aula 07: Visualizado dia 16/10/2020**

***Operadores do JS(parte 1)***

Tipos de operadores: Aritméticos, Atribuição. Relacionais, Lógicos, Ternário.

O que será visto nessa Aula:

Aritméticos e Atribuição

**Aritméticos**

“+” : soma EX: 5 + 2 = 7

“-” : subtração EX: 5 – 2 = 3

“\*” : Multiplicação EX: 5 \* 2 = 10

“/” : Divisão ‘real’ EX: 5 / 2 = 2.5

“%”: Resto da Divisão real EX: 5 % 2 = 1

“\*\*” : Potência EX: 5 \*\* 2 = 25

**Ordem de precedência:**

A seguir está a ordem que deve ser priorizada na hora de utilizar operações.

( ) 🡪 \*\* 🡪 \* , / , % 🡪 + , - .

.:Caso tenha \* ou / ou % na mesma equação é priorizado o que aparece primeiro.

**Incremento:**

var x = 5

x = x + 1 x+=1

x = x – 1 x -=1

Utilizando o operador de incremento essa operação fica:

x++

x--

**Atribuição**

**Atribuição Simples:**

Exemplos:

.:Lembrando que o “=” significa “recebe” (atribui).

var a = 5 + 3 R: 8

var b = a % 5 R: 3

var c = 5 \* b \*\* 2 R: 45

var d = 10 – a /2 R:6

var e = 6 \* 2/d R:2

**Auto atribuições:** São atribuições a própria variável

.:Em JS também podemos usar a sintaxe simplificada

var n = 3 R: 3

n = n + 4 ou n +=4 R: 7 (n deixou de valer 3 agora n = 7 )

n = n – 5 ou n -= 5 R: 2

n = n \* 4 ou n \*= 4 R: 8

n = n / 2 ou n /= 2 R: 4

n = n \*\* 2 ou n \*\*= 2 R: 16

n = n % 5 ou n %= 5 R: 1

No fim a variável “n” que começou valendo 3 terminou valendo 1

**Aula 08: Visualizado dia 18/10/2020**

*Operadores (parte 2)*

O que será visto nessa Aula:

Relacionais, lógicos, ternário.

**Relacionais**

.: Para toda expressão que tenha uma operação relacional ligado a ela, o resultado da operação será sempre um valor booleano (verdadeiro ou falso).

“>” : maior EX: 5 > 2 🡪 true

“<” : menor EX: 7 < 4 🡪 false

“>=” : maior ou igual EX: 8 >= 8 🡪 true

“<=” : menor ou igual EX: 9 <= 7 🡪 false

“==” : igual EX: 5 == 5 🡪 true

“!=” : diferente EX: 4 != 4 🡪 false

**Identidade:**

.: O sinal de igualdade não diferencia o tipo.

5 == 5 🡪 true

5 == ‘5’ 🡪 true

No JavaScript existe um operador de identidade.

5 === ‘5’ 🡪 false

“ ===” : idêntico

No qual essa identidade diz indica se eles são idênticos.

5 !== ‘5’ 🡪 true

“!== ” : desigual restrito. Indica se eles são de tipos diferentes.

**Lógicos**

“!” : Negação .

“&&” : Conjunção (seria o “e” logico).

“||”: Disjunção (seria o “ou” logico).

*Exemplo de negação:*

!true 🡪 false //No caso estaria negando o verdadeiro transformando-o em falso

!false 🡪 true // No caso estaria negando o falso transformando-o em verdadeiro

*Exemplo de Conjunção:*

Operador binário

true && true 🡪 true.

true && false 🡪 false.

false && true 🡪 false.

false && false 🡪 false.

*Exemplo de Disjunção:*

true || true 🡪 true.

true || false 🡪 true.

false || true 🡪 true.

false || false 🡪 false.

**Ordem de precedência:**

Operadores aritméticos 🡪 operadores relacionais 🡪 operadores lógicos.

(), \*\*, / ou \* ou %, + ou - 🡪 n tem ordem específica 🡪 ! , &&, ||.

**Ternário**

Ele junta 3 operandos.

teste ? true : false

//teste: é o teste logico; true: o que irá acontecer quando o teste for verdadeiro; false: o que iria acontecer quando o teste for falso.

*Exemplo*

média >= 7.0 ? “Aprovado” : “Reprovado”

MODULO **B** FINALIZADO!!!

INICIANDO O MODULO **C**!!!

**Aula 09: Visualizado dia 22/10/2020**

*Introdução ao DOM*

Entendendo o DOM.

**O que é DOM?**

É um acrônimo de:

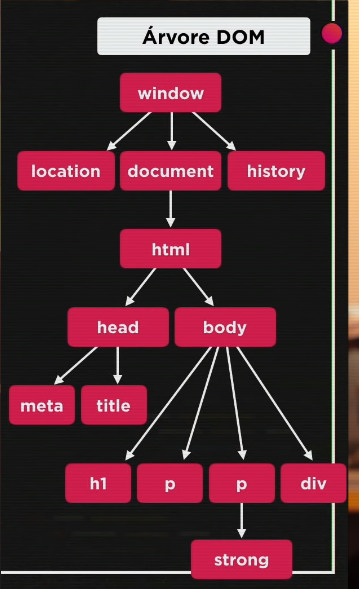
* Document
* Object
* Model

*“Modelo de objetos para documentos.”*

Basicamente é um conjunto de objetos dentro do navegador que fornece acesso aos componentes internos do web site.

Ele está presente quando você utiliza JS no navegador.

**Arvore DOM**



Tudo no javascript está dentro de um objeto window

Dentro do window temos vários outros objetos como:

* Location: localização do site url.
* Document: documento atual.
* History: histórico.

Dentro do document temos o *objeto* html.

Dentro de html temos 2 filhos que seriam o *head e body*.

Html é *“parent”* do head e body, mas Html é *“child”* do document.

Basicamente é uma posição de hierarquia.

*Elemento:* Todos os retângulos na arvore DOM.

**Selecionando o Elemento na arvore DOM.**

**Por Marca:** //tag name

Comando:

getElementsByTagName() //Aviso: você consegue selecionar mais de um objeto com esse comando

**Por ID:**

Geralmente utilizado quando o site é mais evoluído

Comando:

getElementsById()

**Por Nome:**

Também pode usar o nome do objeto se você tiver a propriedade name.

Comando:

getElementsByName()

**Por Classe:**

Comando:

getElementsByClassName()

**Por Seletor:**

Comando:

Quando é ID “#”.

Quando é Class “.”.

querySelector()

querySelectorAll()

Exemplo: var d = window.document.querySelector('div**#**msg')